AMENDMENTS TO THE CLAIMS

Please cancel claims 10-14 and 41-45 without prejudice. Kindly amend claims 1-5, 7-9, 15-16, 32-36, 38-39, 40, and 46 as shown in the following listing of claims. The listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims

1. (Currently Amended) An apparatus in a pipeline microprocessor, for ensuring coherency of instructions within stages of the pipeline microprocessor, the apparatus comprising:

instruction cache management logic, configured to receive an address corresponding to a next instruction to be fetched, and configured to detect that a part of a memory page corresponding to said next instruction to be fetched cannot be freely accessed without checking for coherency of the instructions within said part of said memory page and, upon detection, configured to provide said address, wherein said instruction cache management logic evaluates an instruction translation lookaside buffer (ITLB) entry corresponding to said address to detect that said part cannot be freely accessed; and

synchronization logic, configured to receive said address from said instruction cache management logic, and configured to direct data cache management logic to check for coherency of the instructions within said part of said memory page, and, if the instructions are not coherent within said part of said memory page, said synchronization logic is configured to direct the pipeline microprocessor to stall a fetch of said next instruction to be fetched until the stages of the pipeline microprocessor have executed all preceding instructions, wherein said data cache management logic evaluates a data translation lookaside buffer (DTLB) entry corresponding to said address to detect that the instructions are not coherent within said part of said memory page.

2. (Original) The apparatus as recited in claim 1, wherein if said instruction ~~said instruction~~ cache management logic ~~evaluates an instruction translation lookaside buffer (ITLB) entry corresponding to said address to detect~~detects that said part cannot be freely accessed, regardless of whether said instructions are coherent or not, said DTLB entry contents are copied into said ITLB entry by said synchronization logic so that an address translation lookup sequence is avoided.

3. (Currently Amended) The apparatus as recited ~~in claim 2~~in claim 1, wherein said ITLB entry and said DTLB entry both ~~corresponds~~ correspond to said memory page.

4. (Currently Amended) The apparatus as recited in claim 3, wherein said ITLB entry and said DTLB entry each comprises a plurality of part-page ownership bits.

5. (Currently Amended) The apparatus as recited in claim 4, wherein a first one of said plurality of part-page ownership bits in said ITLB entry and a second one of said plurality of part-page ownership bits in said DTLB entry both correspond~~corresponds~~ to said part of said memory page.

6. (Original) The apparatus as recited in claim 5, wherein remaining ones of said plurality of part-page ownership bits correspond to remaining parts of said memory page.

7. (Currently Amended) The apparatus as recited in claim 5, wherein said part can be freely accessed if first said one of said plurality of part-page ownership bits is set in said ITLB entry.

8. (Currently Amended) The apparatus as recited in claim 5, wherein said part cannot be freely accessed if said first one of said plurality of part-page ownership bits is not set in said ITLB entry.

9. (Original) The apparatus as recited in claim 4, wherein said plurality of part-page ownership bits comprise four part-page ownership bits each in said ITLB and DTLB entries, and wherein said part comprises one-quarter of said memory page.

10. (Cancelled)

11. (Cancelled)

12. (Cancelled)

13. (Cancelled)

14. (Cancelled)

15. (Currently Amended) The apparatus as recited in ~~claim 13~~claim 5, wherein the instructions are not coherent within said part if said ~~one of~~second one of said plurality of part-page ownership bits is set.

16. (Original) The apparatus as recited in ~~claim 13~~claim 5, wherein the instructions are coherent within said part if said one of said plurality of part-page ownership bits is not set.

17. (Cancelled)

18. (Cancelled)

19. (Cancelled)

20. (Cancelled)

21. (Cancelled)

22. (Cancelled)

23. (Cancelled)

24. (Cancelled)

25. (Cancelled)

26. (Cancelled)

27. (Cancelled)

28. (Cancelled)

29. (Cancelled)

30. (Cancelled)

31. (Cancelled)

32. (Currently Amended) A method in a pipeline microprocessor, for ensuring coherency of instructions within stages of the pipeline microprocessor, the method comprising:

~~within~~ within an instruction cache, detecting that a part of a memory page corresponding to a next instruction to be fetched cannot be freely accessed without checking for coherency of the instructions within the part of the memory page, wherein said detecting comprises:

evaluating an instruction translation lookaside buffer (ITLB) entry corresponding to an address for the next instruction;

directing logic within a data cache to check for coherency of the instructions within the part of the memory page, wherein said directing comprises

evaluating a data translation lookaside buffer (DTLB) entry corresponding to the address for the next instruction; and

if the instructions are not coherent, stalling a fetch of the next instruction from the instruction cache until the stages of the pipeline microprocessor have executed all preceding instructions.

33. (Currently Amended) The method as recited in claim 32, ~~wherein said detecting comprises~~further comprising:

regardless of whether said instructions are coherent or not, if the part cannot be freely accessed, copying the DTLB entry contents into the ITLB entry, whereby an address translation lookup sequence is avoided.~~evaluating an instruction translation lookaside buffer (ITLB) entry corresponding to an address for the next instruction.~~

34. (Currently Amended) The method as recited in claim 33, wherein the ITLB entry and the DTLB entry both correspond~~corresponds~~ to the memory page.

35. (Currently Amended) The method as recited in claim 34, wherein the ITLB entry and the DTLB entry each comprises a plurality of part-page ownership bits.

36. (Currently Amended) The method as recited in claim 35, wherein a first one of the plurality of part-page ownership bits in the ITLB and a second one of the plurality of part-page ownership bits in the DTLB both correspond~~corresponds~~ to the part of the memory page.

37. (Original) The method as recited in claim 36, wherein remaining ones of the plurality of part-page ownership bits correspond to remaining parts of the memory page.

38. (Original) The method as recited in claim 37, wherein the part of said memory page can be freely accessed if the first one of the plurality of part-page ownership bits is set.

39. (Currently Amended) The method as recited in claim 37, wherein the part of the memory page cannot be freely accessed if the first one of the plurality of part-page ownership bits is not set.

40. (Currently Amended) The method as recited in claim 35, wherein the plurality of part-page ownership bits comprise four part-page ownership bits each in the ITLB and DTLB entries, and wherein the part comprises one-quarter of the memory page.

41. (Cancelled)

42. (Cancelled)

43. (Cancelled)

44. (Cancelled)

45. (Cancelled)

46. (Currently Amended) The method as recited ~~in claim 45~~in claim 35, wherein the instructions are not coherent within the part if the second one of ~~said plurality~~ the plurality of part-page ownership bits is set.